UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/637,219 | 08/08/2003 | Benoit Maison | YOR920030225US1 | 3318 |

35526 7590 04/21/2008

DUKE W. YEE
YEE & ASSOCIATES, P.C.
P.O. BOX 802333
DALLAS, TX 75380

| EXAMINER | |
|---|---|
| NEWAY, SAMUEL G | |
| ART UNIT | PAPER NUMBER |
| 2626 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 04/21/2008 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

Application Number: 10/637,219
Filing Date: August 08, 2003
Appellant(s): MAISON ET AL.

Theodore D. Fay III
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 06 February 2008 appealing from the Office action mailed 17 July 2007.

**(1) Real Party in Interest**

A statement identifying by name the real party in interest is contained in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The statement of the status of claims contained in the brief is correct.

**(4) Status of Amendments After Final**

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

**(5) Summary of Claimed Subject Matter**

The summary of claimed subject matter contained in the brief is correct.

**(6) Grounds of Rejection to be Reviewed on Appeal**

**WITHDRAWN REJECTIONS**

The following grounds of rejection are not presented for review on appeal because they have been withdrawn by the examiner. The 35 USC § 101 Rejections of claims 11 – 16 are withdrawn.

**(7) Claims Appendix**

The copy of the appealed claims contained in the Appendix to the brief is correct.

**(8) Evidence Relied Upon**

Arnold et al. US PGPub 2003/0125955, December 28, 2001

| | | |
|---|---|---|
| 6,321,372 | Poirier et al. | 12-1998 |
| 5787285 | Lanning | 08-1995 |

IBM Technical Disclosure Bulletin, "Determining the Probability of Words in a String with a Word-Skipping Model", Nov 1, 1985

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

***Claim Rejections - 35 USC § 103***

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1 – 4, 6 – 7, 11 – 14, and 21 – 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Arnold et al. (USPGPub 2003/0125955) in view of Poirier et al (USPN 6,321,372).

Claim 1:

Arnold discloses a method, implemented in a data processing system, for generating task-specific code for pattern recognition ([0008]), the method comprising:

receiving task-specific input system data of a pattern recognition system and generating task-specific code for the pattern recognition system based on the task-specific input system data ("the distributed speech recognition system allows automatic "speaker adaptation" to be performed locally by the client device ... local parameters ... are adapted locally by the client device in performing its speech recognition function", [0008], see also [0009] and "the client device is adapting its models in response to the speaker", [0020]).

However, even if Arnold discloses that "the speech recognizer module ... can be represented by one or more software applications" [0054] where the speech recognizer includes speaker and noise adaptation modules which perform the "speaker adaptation", it does not explicitly disclose these modules as software source code.

Poirier discloses a similar method where a source code is modified, such as by further specifying it, in a linguistic service system. Poirier also discloses the code being compiled ("compile ... modified source code", col. 10, lines 52-56).

It would have been obvious to one with ordinary skill in the art at the time the invention was made to use a source code (which has to be compiled in order to execute

on a computer) as the software modules in Arnold's method in order to modify an existent source code in order to generate a more specified service (Poirier, col. 2, lines 26-36).

Claim 2:

Arnold and Poirier disclose the method of claim 1, Arnold further discloses wherein the pattern recognition system performs speech recognition ("parameters ... are adapted locally by the client device in performing its speech recognition functions", [0008]).

Claim 3:

Arnold and Poirier disclose the method of claim 2, Arnold further discloses wherein the task-specific input system data includes one of a language model, an acoustic model, a front-end for computing feature vectors, and information related to speaker adaptation (FIG. 1, items 120 – 124 and related text).

Claim 4:

Arnold and Poirier disclose the method of claim 3, Arnold further discloses wherein the acoustic model includes Gaussians ([0023]).

Claim 6:

Arnold and Poirier disclose the method of claim 3, Arnold further discloses wherein the acoustic model is represented as a Hidden Markov Model ([0023]).

Claim 7:

Arnold and Poirier disclose the method of claim 1. Poirier further discloses the source code being compiled ("compile ... modified source code", col. 10, lines 52-56).

It would have been obvious to one with ordinary skill in the art at the time the invention was made to compile a source code in order to obtain an executable file which is able to run on a computer (Poirier, "compile ... modified source code to obtain service executable", col. 10, lines 52-56).

Claims 11 – 13 are similar in scope and content to claims 1 – 3 and are rejected with the same rationale.

Claim 14 is similar in scope and content to claim 7 and is rejected with the same rationale.

Claim 21:

Arnold discloses an apparatus for generating task-specific code for pattern recognition ([0008]), the method comprising:

a bus; a memory connected to the bus, wherein the memory contains computer readable instructions; a processor connected to the bus, wherein the processor executes the computer readable instructions (FIG. 2 and related text) to:

receive task-specific input system data of a pattern recognition system and generate task-specific code for the pattern recognition system based on the task-specific input system data ("the distributed speech recognition system allows automatic "speaker adaptation" to be performed locally by the client device ... local parameters ... are adapted locally by the client device in performing its speech recognition function", [0008], see also [0009] and "the client device is adapting its models in response to the speaker", [0020]).

However, even if Arnold discloses that "the speech recognizer module ... can be represented by one or more software applications", it does not explicitly disclose the software applications being source code.

Poirier discloses a similar method where a source code is modified, such as by further specifying it, in a linguistic service system. Poirier also discloses the code being compiled ("compile ... modified source code", col. 10, lines 52-56).

It would have been obvious to one with ordinary skill in the art at the time the invention was made to use a source code (which has to be compiled in order to execute on a computer) as the software modules in Arnold's apparatus in order to modify an existent source code in order to generate a more specified service (Poirier, col. 2, lines 26-36).

Claim 22 is similar in scope and content to claim 2 and is rejected with the same rationale.

Claim 23 is similar in scope and content to claim 23 and is rejected with the same rationale.

3. Claim 5 is rejected under 35 U.S.C. 103(a) as being unpatentable over Arnold et al. (USPGPub 2003/0125955) in view of Poirier et al (USPN 6,321,372) and in further view of ("Determining the Probability of Words in a String With a Word-Skipping Model ", IBM Technical Disclosure Bulletin, November 1985).

Claim 5:

Arnold and Poirier disclose the method of claim 3, but they do not disclose wherein the language model is represented as a Hidden Markov Model ([0023]).

The IBM Technical Disclosure Bulletin discloses a speech recognition method where the language model is "defined as a Markov source (a hidden Markov chain)" (page 2, lines 11-13).

Therefore it would have been obvious to one with ordinary skill in the art at the time of the invention to represent the language model as a Hidden Markov Model in Arnold's method because off the shelf Hidden Markov Model software was available therefore freeing Arnold from programming another model.

4. Claims 8 – 10, 15 – 16, and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Arnold et al. (USPGPub 2003/0125955) in view of Poirier et al (USPN 6,321,372) and in further view Lanning (USPN 5,787,285).

Claim 8:

Arnold and Poirier disclose the method of claim 7, but they do not explicitly disclose profiling the decoder program to form a profile and determining whether the decoder program is optimized.

Lanning discloses a method of optimizing executable software where the code is profiled and optimized ("automated "profilers" to provide data to these optimizing compilers", col. 1, lines 42-52).

It would have been obvious to one with ordinary skill in the art at the time the invention was made to profile and optimize the code in Arnold and Poirier's method in order to "enhance the run-time performance of a piece of software"(Lanning, col. 1, lines 42-52).

Claim 9:

Arnold, Poirier and Lanning disclose the method of claim 8, Arnold and Poirier do not explicitly disclose wherein responsive to the decoder program not being optimized, automatically modifying and recompiling the decoder program based on the profile.

Lanning discloses automatically modifying and recompiling the code ("The optimizing compiler then uses the information gathered by the profiler to recompile the source code", col. 1, lines 42-52).

It would have been obvious to one with ordinary skill in the art at the time the invention was made to modify and recompile the code in Arnold and Poirier's method in order to "enhance the run-time performance of a piece of software"(Lanning, col. 1, lines 42-52).

Claim 10:

Arnold and Poirier disclose the method of claim 7, but they do not disclose compiling the code in several parts corresponding to several modules of the pattern recognition system and assembling the compiled code before execution.

Lanning discloses compiling the code in several parts as claimed in the instant claims (col. 3, lines 33-42).

It would have been obvious to one with ordinary skill in the art at the time the invention was made to compile the code in several parts in Arnold and Poirier's method because this "reduces the execution time of the code associated with that environment or mode" (Lanning, col. 3, lines 51-55).

Claims 15 – 16 are similar in scope and content to claims 8 – 9 and are rejected with the same rationale.

Claim 24 is similar in scope and content to claim 8 and is rejected with the same rationale.

**(10) Response to Argument**

Appellant's arguments filed 06 February 2008 have been fully considered but they are not persuasive.

1. Regarding claims 1 – 4, 6 – 7, 11 – 14, and 21 – 23:

Appellant argues that the combination of Arnold and Poirier fails to teach the limitation in claim 1 which states "generating task-specific code for the pattern recognition system based on the task-specific input system data, wherein the task-specific code includes computer language suitable for compilation" (Appeal Brief, p. 11, paragraph 5) because "Arnold specifically teaches away from generating code for a client in response to a client request as claimed" (Appeal Brief, p. 13, paragraph 3). It is noted that claim 1 generates code based on task-specific input system data (for example a language model, an acoustic model, or information related to speaker adaptation) but not in response to a client request as is stated.

Appellant further quotes Arnold as stating in paragraph [0020] "… ready to use models, portions or updates". This quote is inaccurate as Arnold never explicitly cites 'ready to use models, portions or updates' in [0020] or anywhere else in the specification. In [0020] Arnold discloses employing ""dynamic grammars" as a driving mechanism in providing the necessary models or portions or updates of a model to the client device". The server picks only the necessary models (language models) to send

to the client and the client uses these models to update or transform its existing models, this is similar to Appellant's generating a code based on a task-specific input system data (such as language models).

Appellant has also submitted that adding the teaching of Poirier to Arnold would cause Arnold to become inoperative because adding "the teaching of Poirier to that of Arnold would cause Arnold to become inoperative because a request from a client would no longer be ready to use and require compilation, thereby eviscerating the current method of Arnold" as Arnold teaches away from heavy processing on the client computer (Appeal Brief, p. 14, paragraph 1). However this argument is not persuasive at least because Arnold discloses that the server and the client could be each implemented using "a general purpose computer". Thus any processing that can be performed on a general purpose computer such as compiling and executing source code could be performed on Arnold's client device.

Appellant also argues that there is no proper reason given by the Examiner "to achieve the legal conclusion of obviousness of claim 1 under the standards of KSR Intl." (Appeal Brief, p. 14, paragraph 3). The Examiner respectfully disagrees because a teaching disclosed in Poirier (col. 2, lines 26-36) to combine the two references is given as satisfying the basic requirements of a prima facie case of obviousness under KSR. Poirier discloses that a less specified linguistic service with a preexisting source code (similar to Arnold's not yet updated language model) is modified to produce a new source code for a more specified linguistic service (similar to Arnold's updated language

model). This teaching in Poirier would have led one with ordinary skill in the art at the time of the invention to combine Arnold and Poirier to arrive at the claimed invention.

2. Regarding claim 5:

Appellant only states that the IBM Bulletin fails to remedy the issues raised regarding claims 1 – 4, 6 – 7, 11 – 14, and 21 – 23.

3. Regarding claims 8 – 10, 15 – 16, and 24:

Appellant also argues that Lanning fails to teach profiling a source code and determining whether the source code is optimized. However, Lanning discloses software developers "tuning" their code for a particular operation (col. 1, lines 24-31). Since there will be no need to optimize the software if it is already tuned to a particular operation, "tuning" implies some kind of checking mechanism to verify if the software has already been optimized. As to the argument that "Lanning does not teach creating profiles from the code to be tested as is currently claimed", Appellant is equating Lanning's "profile" to the profile as disclosed in Appellant's specification. However Lanning's profile actually reads on Appellant's "typical input parameters" (Appellant specification, page 5, lines 10-12) and the element that read on Appellant's profile is Lanning's run-time information which is the result of the profiling process (Lanning, col. 3, lines 1-10). Lanning disclosure of profiling a source code and forming the run-time information reads on Appellant's limitation of "profiling the decoder program to form a profile" because Appellant's profile is disclosed as the result of the profiling performed

on the code ("a program is first written and run with typical input parameters, and an execution profile is generated. This profile indicates such events as cache-misses ... ", Appellant specification, page 5, lines 10-14).

**(11) Related Proceeding(s) Appendix**

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Samuel G Neway/
Samuel G. Neway

Conferees:

/Talivaldis Ivars Smits/
Primary Examiner, Art Unit 2626
Talivaldis Smits

/David R Hudspeth/
Supervisory Patent Examiner, Art Unit 2626

4/16/2008